

---

# Segmentation of Nerve Bundles and Ganglia in Spine MRI using Particle Filters

by

Adrian Vasile Dalca

B.S., Computer Science and Physics, University of Toronto, 2008

M.S., Computer Science, University of Toronto, 2010

---

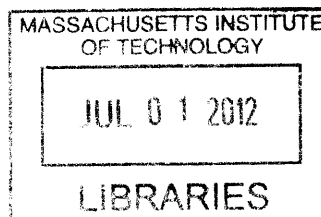
Submitted to the Department of Electrical Engineering and Computer Science  
in partial fulfillment of the requirements for the degree of

Master of Science  
in Electrical Engineering and Computer Science  
at the Massachusetts Institute of Technology

June 2012

© 2012 Massachusetts Institute of Technology  
All Rights Reserved.

**ARCHIVES**



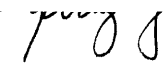
Signature of Author: \_\_\_\_\_

Adrian Vasile Dalca  
Department of Electrical Engineering and Computer Science  
May 23, 2012

Certified by: \_\_\_\_\_

Polina Golland  
Associate Professor of Electrical Engineering and Computer Science  
Thesis Supervisor

Accepted by: \_\_\_\_\_

 Leslie A. Kolodziejski  
Professor of Electrical Engineering and Computer Science  
Chair, Committee for Graduate Students



---

---

## Segmentation of Nerve Bundles and Ganglia in Spine MRI using Particle Filters

by Adrian Vasile Dalca

Submitted to the Department of Electrical Engineering and Computer Science  
in partial fulfillment of the requirements for the degree of  
Master of Science

### **Abstract**

Automatic segmentation of spinal nerve bundles originating within the dural sac and exiting the spinal canal is important for diagnosis and surgical planning. The variability in intensity, contrast, shape and direction of nerves seen in high resolution myelographic MR images makes segmentation a challenging task. In this thesis, we present an automatic tracking method for segmentation of nerve bundles based on particle filters.

We develop a novel approach to flexible particle representation of tubular structures based on Bézier splines. We construct an appropriate dynamics to reflect the continuity and smoothness properties of real nerve bundles. Moreover, we introduce a robust image likelihood model that enables delineation of nerve bundles and ganglia from the surrounding anatomical structures. We evaluate the results by comparing them to expert manual segmentation, and we demonstrate accurate and fast nerve tracking.

---

Thesis Supervisor: Polina Golland

Title: Associate Professor of Electrical Engineering and Computer Science



---

---

# Acknowledgments

I would like to thank my supervisor, Prof. Polina Golland, for her exceptional guidance throughout this work. Her support and devotion to her students is truly rare, and her ideas, feedback and approach to research have been the guiding forces in developing my project and increasing my potential. I thank her for keeping me focused and providing a stimulating and challenging environment to which I undoubtedly owe my current understanding of research. She has given me considerable insight into how to approach problems, what is important in my research and how to challenge myself. Through it all she remains friendly and supportive, and I am greatly in her debt.

A heartfelt and ongoing thank you goes to my fiancée and life partner, Monica Stanciu for always being so incredibly wonderful to me in all aspects of life. I owe a great deal of who I am today to her. Her constant, cute enthusiasm keeps me happy, sane and optimistic day-to-day. Her admiration for science and her remarkable self-discipline keeps me focused and motivated in a sea of distractions. Most of all, her seemingly infinite love and support make me better in every way I can imagine. Thank you.

My parents, Ana-Maria and Adrian, and dear sister Ioana have been a beacon of support throughout my life. Their love, guidance and tremendous sacrifices have carved my path for most of my life and have helped lead me to where I am. Their encouragement has driven me to pursue my potential, and I am truly grateful.

I've been very fortunate to have an extremely supportive, fun and productive research environment. A big thank you goes to the members of the Medical Vision Group: Ramesh Sridharan, Andy Sweet, George Chen, Archana Venkataraman, Christian Wachinger, Danial Lashkari, Michal Depa, Bjoern Menze and Georg Langs, for all the fun, support and help throughout my time here so far. I am further grateful to everyone in the Vision-Graphics Neighbourhood for an incredibly stimulating and friendly research community. I would also like to thank everyone who has given me feedback, comments, and ideas throughout this project, especially Prof. Ehud Schmidt, for his specific clinical guidance, motivation and data that have made this project possible.



---

---

# Contents

<b>Abstract</b>	<b>3</b>
<b>Acknowledgments</b>	<b>4</b>
<b>List of Figures</b>	<b>9</b>
<b>1 Introduction</b>	<b>13</b>
1.1 Background . . . . .	13
1.2 Contributions . . . . .	15
1.3 Thesis Outline . . . . .	16
<b>2 Prior Work</b>	<b>17</b>
2.1 Vessel Segmentation . . . . .	17
2.2 Other Related Work . . . . .	18
<b>3 Particle Filter</b>	<b>19</b>
3.1 The Particle Filter . . . . .	19
3.2 Algorithms . . . . .	20
<b>4 Nerve Segmentation</b>	<b>23</b>
4.1 Particle Representation for Nerve Tracks . . . . .	23
4.2 Dynamics model . . . . .	26
4.3 Likelihood Measure . . . . .	27
<b>5 Implementation Details and Parameter Choices</b>	<b>31</b>
<b>6 Results</b>	<b>33</b>
6.1 Synthetic Data . . . . .	33

---

6.2 Patient Data . . . . .	35
<b>7 Discussion</b>	<b>37</b>
7.1 Algorithm Behavior . . . . .	37
7.2 Exceptions . . . . .	38
<b>8 Conclusion</b>	<b>39</b>
8.1 Contributions . . . . .	39
8.2 Future Work . . . . .	39
<b>Bibliography</b>	<b>41</b>



---

---

# List of Figures

1.1	Nerve bundles (yellow) group inside the vertebral canal, and form ganglia while exiting through the foramen (spacing between the vertebra). Visualization done with Zygote Body software [6] . . . . .	14
1.2	Different slices from an example MRI of a patient with a herniated disk. The nerve bundles appear dark gray inside the spinal canal, with one bundle highlighted via a blue overlay in the top left. All arrows point to the same nerve bundle. Blue arrows show examples of poor contrast between the nerve and surrounding tissue; orange arrows indicate the thickening of the nerve into a ganglion. The purple circle shows an area of pathology, where the (dark) disk is protruding into the spinal canal and putting pressure on the nerve bundle. . . . .	15
4.1	An overview of the Particle Filter algorithm as applied to nerve segmentation. To best illustrate the concepts, we use very few particles in our illustrations, unlike the actual implementation where several thousand particles are used at each step. Following the initialization, we perform the illustrated procedure iteratively. The previous posterior distribution $p(\mathbf{h}_{t-1} \mathbf{z}_{t-1})$ is approximated by weighed particles (nerve segments). In the first step of an iteration, we obtain samples (orange dots) based on this posterior. In the second step, we propagate these samples, i.e., continue the previous nerve segments, via the dynamics model. Finally, the new particles are weighed based on the current observation with the likelihood model. . . . .	24

- 4.2 Left: Nerve segment (particle) is defined as a Bézier curve centerline with four control points, a quadratic radius function  $r(\cdot)$ , and the average intensity. Right: An observation is a sub-volume of the entire MRI volume, surrounding a particle hypothesis. . . . . 25
- 4.3 Illustration of likelihood computation and the partial volume effect. Left: hypothesized particle implies gradient directions (blue arrows) in the image. Next to it are two examples of possible observations around this hypothesis. In the first example observation, the proposed intensity is very different from the average intensity over the particle volume, giving rise to high intensity distance  $d_\mu^2$ . The gradient directions at each pixel are also very different from the expected directions (at the same voxel), yielding a high gradient distance  $d_\nabla^2$  for the first example observation. For the second observed image, both the average intensity over the particle volume and the gradient directions match the expected particle significantly better. Overall, the particle weight will be low given the first observation, but high given the second observation. Right: a thin particle, or nerve segment, is shown where the blue grid defines voxels. Only the shaded voxels are almost entirely in the nerve, the rest are a combination of nerve and background intensities, illustrating the severity of the partial volume effect in our application. . . . . 28
- 6.1 Synthetic nerves generated from our model with added white noise embedded in different backgrounds: blank, white noise, Perlin noise and anatomical patch, respectively. The yellow outlines show the automatic segmentation results. Slight under-segmentations can be seen around the edges, and occasionally at the ends of the nerve section, where the segmentation has moved into the adjoining slices one voxel too early. . . 33
- 6.2 Real nerve delineation with added white noise and embedded in different backgrounds: blank, white noise, Perlin noise and anatomical patch, respectively. The yellow outlines show automatic segmentations. The segmentations tend to avoid irregularities in the nerves, and will usually under-segment. For such real nerves, note that the segmentations are initialized at the top of the nerves in the shown slice, which leads to under-segmentations in the region due to the initializations. . . . . 33

- 
- 6.3 Summary of synthetic results for the eight types of synthetic nerves and backgrounds. The red bars indicate the average of the median distance between the automatic and the manual outline. The blue bars indicate the 90<sup>th</sup> percentile. For each category, *Model* indicates that we generated the nerve via our particle model, and *Expert* indicates nerves generated from the smoothed expert segmentation of real nerves. . . . . 34
- 6.4 Patient data results. Left: Rendered segmentation results – a herniated disk (manually segmented, yellow) is impinging on the nerve tracts (segmented with our algorithm with minimal user input, green). Right: Slice that includes a section of a nerve and a ganglion. The yellow outline indicates the automatic delineation of the nerve. The algorithm tends to slightly under-segment the nerve, as we see the margins of the nerve underneath the segmentation. . . . . 35
- 6.5 Summary of the 12 segmentation results on real patient data. The red bars indicate the medians of distance between the automatic and the expert segmentation; the blue bars indicate the 90<sup>th</sup> percentile, in voxels. 35



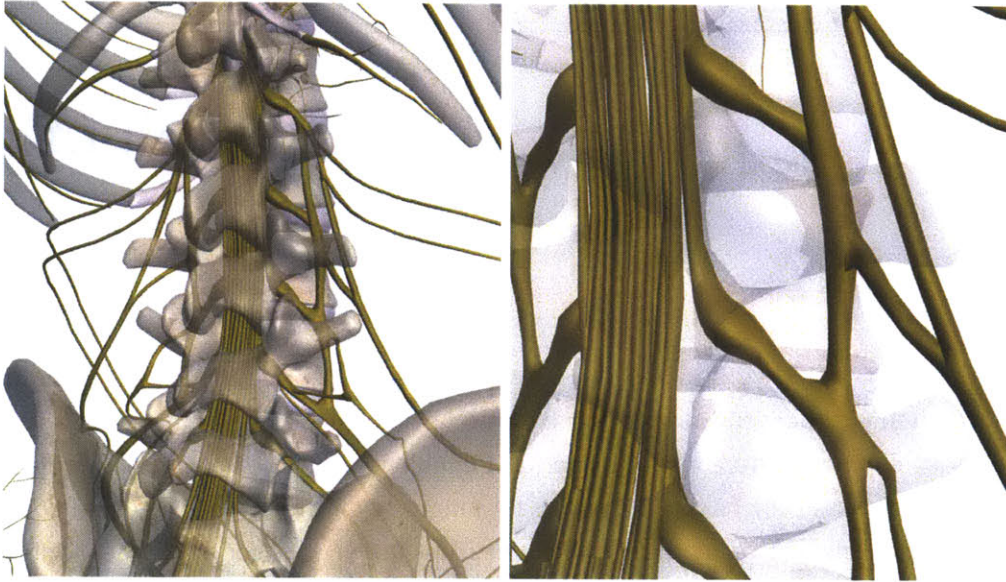
# Introduction

**M**APPING of nerve bundles is essential for diagnosis and treatment planning of spinal pathologies, and offers substantial benefits for image-guided interventions. Recent developments in high-resolution MRI enable visualization of nerve bundles from within the dura, as they pass through the foramen and exit the vertebral canal [24]. The bundles are characterized by dark gray intensities, which offer good contrast with white fluids and black bone, but are often of similar intensity to that of marrow and muscle. Manual segmentation of nerves and ganglia in these 3D volumes is quite challenging and time-consuming. In this thesis, we propose and demonstrate a method for automatic segmentation of nerve bundles and ganglia in high-resolution MRI that requires minimal input from an expert.

## ■ 1.1 Background

Spinal nerves are cable-like structures of axons that carry autonomic signals between the brain and the rest of the body. All spinal nerves eventually emerge from the vertebral canal through an opening, or foramen. The nerve ganglia, which contain the nerve cell bodies, are formed during the passage (Figure 1.1). The degeneration and herniation of spinal disks leads to pressure or pinching of the surrounding regions, including spinal nerves and nerve ganglia, causing pain in the neck, back or extremities. In extreme cases this can lead to more damaging changes, such as paralysis.

An understanding of where and how nerves are affected is crucial for diagnosis and treatment planning. Minimally invasive surgery is often the preferred treatment, however, the required small incisions highly limit the visual field of the surgeon. Nerve segmentation, along with derived measures or properties, can help to visualize the pathologies and thus lead to improved diagnosis. It can also benefit surgery and minimally invasive interventions by reducing procedure duration and complications and by

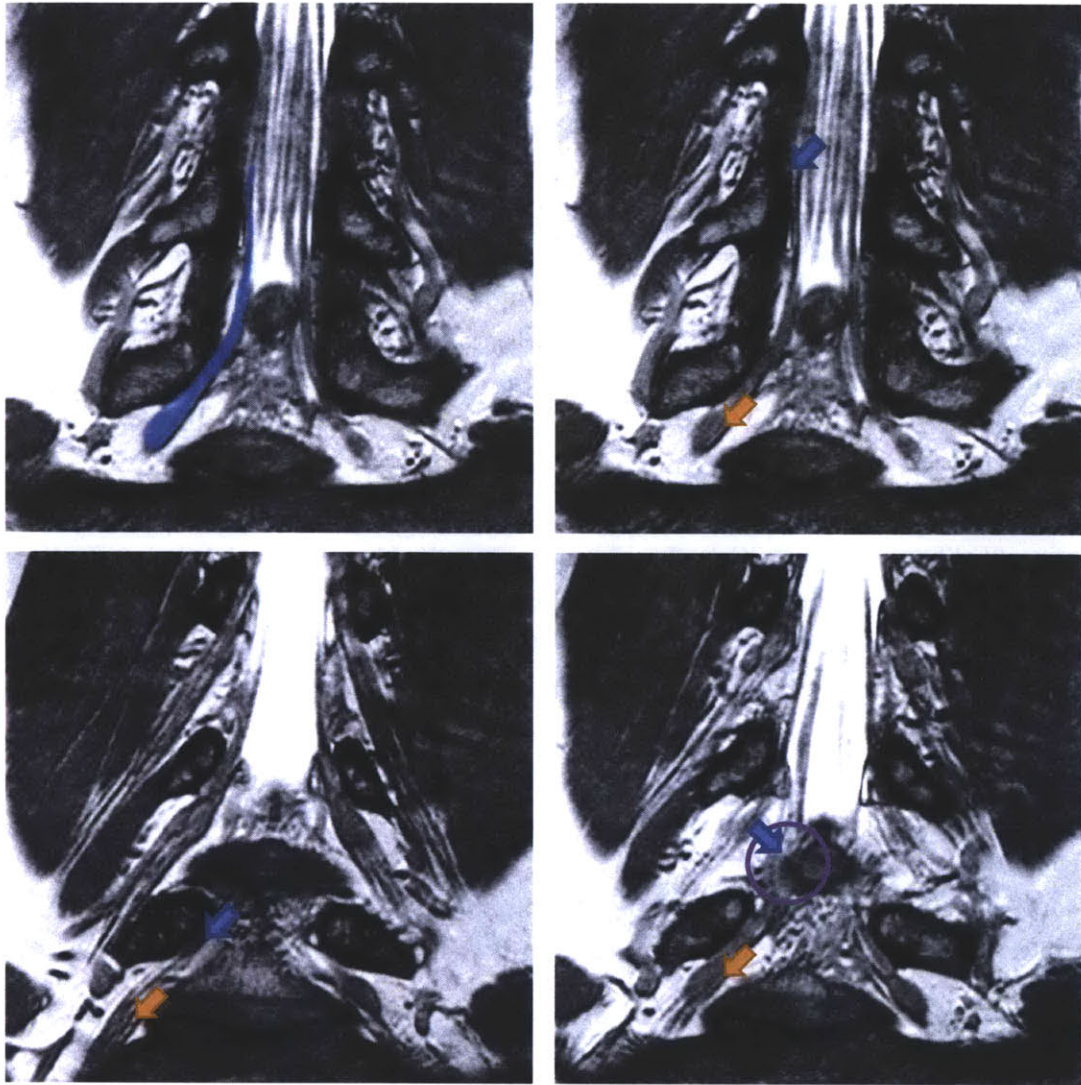


**Figure 1.1.** Nerve bundles (yellow) group inside the vertebral canal, and form ganglia while exiting through the foramen (spacing between the vertebra). Visualization done with Zygote Body software [6]

improving outcomes of treatment of several degenerative conditions and spinal trauma. Manual tracing of the nerves is extremely challenging due to the complex contrast profiles with surrounding anatomy and the 3D nature of the MR volume, where nerves move in and out of slices. The amount of time and effort currently required for manual tracing precludes practical construction of nerve maps. Automated nerve segmentation with minimal user input therefore promises to significantly improve image-based diagnosis, therapy and surgery by providing fast tracing and measurements of the nerve bundles.

Nerve bundles and ganglia can be observed in high-resolution MRI [24], as illustrated in Figure 1.2. The voxels are approximately 0.3-0.6 mm in-plane and 0.9-1 mm thickness. Bundles can be easily seen inside the spinal canal, where they appear as dark gray. The intensity contrast changes along the nerve with neighboring tissues including cerebral spinal fluid, bone, muscle and other nerves, which presents significant challenges for segmentation. The bundles can be thought of as approximately cylindrical structures, but may change shape in the presence of pathologies such as disk herniation (Figure 1.2). As the bundles turn and exit the vertebral canal through the foramen and form nerve ganglia, they grow thicker and less regular in shape. The intensity contrast between the nerve tissue and the surrounding anatomy is further reduced.





**Figure 1.2.** Different slices from an example MRI of a patient with a herniated disk. The nerve bundles appear dark gray inside the spinal canal, with one bundle highlighted via a blue overlay in the top left. All arrows point to the same nerve bundle. Blue arrows show examples of poor contrast between the nerve and surrounding tissue; orange arrows indicate the thickening of the nerve into a ganglion. The purple circle shows an area of pathology, where the (dark) disk is protruding into the spinal canal and putting pressure on the nerve bundle.

## ■ 1.2 Contributions

In this thesis, we present the first method for automatic segmentation of nerves and nerve ganglia in spinal MRI. Specifically, we formulate nerve segmentation as tracking based on particle filtering, also referred to as sequential Monte-Carlo tracking. The particle filter is an ideal approach for our task, as it enables rich hypotheses, flexible dynamics and diverse likelihood models. To address the specific challenges of nerve tracking, we define a flexible particle representation that captures the behaviour of the nerve bundles under both normal conditions and pathologies. We use a Bézier spline [27] centerline with a quadratic radius function to characterize a nerve bundle. We devise a dynamics model for particle updates that enforces continuity and smoothness, and further respects common properties of the nerves. Furthermore, we define an image likelihood measure that compares gradient fields and intensities of predicted patches with image observations to evaluate the posterior distribution of the particles' importance. Once tracking is complete, we remove spurious segmentations by measuring the quality of the entire tract. One additional user click is sometimes required to specify the desired nerve tract among several proposed solutions.

We illustrate our algorithm on various types of synthetic data as well as patient scans. We demonstrate successful segmentations of nerves bundles and ganglia, obtaining accurate estimates of nerve path and thickness. We report example results, and quantitatively compare the surfaces of automatic and desired segmentations, finding strong agreement between the two with a median surface distance of just one voxel.

## ■ 1.3 Thesis Outline

This thesis is organized as follows. Chapter 2 reviews related work. Chapter 3 introduces the theory behind the particle filter. Chapter 4 presents our methodology for nerve tracking in detail. The choices of parameters and implementation details are discussed in Chapter 5. Chapter 6 presents the experimental design, and segmentation results on synthetic and patient data. Chapter 7 offers a discussion of the results and summarizes the strengths and shortcomings of the algorithm. Finally, we conclude with Chapter 8. An initial version of the algorithm [10] was presented at the International Conference on Medical Image Computing and Computer Assisted Intervention.



# Prior Work

**T**O the best of our knowledge, the method we present in this thesis is the first for automatic nerve segmentation. However, closely related prior work has been demonstrated for segmentation of other tubular structures, such as blood vessels. In this chapter, we review the methods used in vessel segmentation and other related fields, and discuss their connection with our nerve segmentation task.

### ■ 2.1 Vessel Segmentation

Region-growing approaches have been used successfully for vessel segmentation. Classical region growing [7], competitive region growing [29] and wave propagation [8, 9] grow an initial point or region to incrementally segment an object based on inclusion criteria for neighboring voxels. The success of these greedy methods depends on good contrast in the vessel images. Unfortunately, the strong contrast required is unavailable for the nerve bundles in the spinal MRI volumes, where nerves may be in contact with many tissues of varying intensities, including other nerve bundles. In these images, region growing methods suffer from severe errors, where segmentation “leaks” into nearby structures in areas that lack contrast [17].

Active contour methods evolve an initial boundary to segment the tubular targets and offer a model-based approach [20, 21]. For example, variational and level-set formulations of these methods have been successfully used for vascular segmentation [11, 20, 22]. However, these methods require time-consuming initialization and suffer from many local minima [17]. Moreover, false positive rate and leakage is certain to limit the usefulness of such methods in our application.

Another approach is to first extract the vessel centerline, and to fill out the segmentation as a subsequent step. The target is modeled as a tubular structure – most often via circular cross-sections of inscribed cylinders or spheres [2, 13]. In practice, these meth-

ods require a fair amount of user interaction to often re-seed tubes or branches [17]. Correction and re-centering during centerline propagation are affected substantially by neighboring tissues of similar intensities. Some methods require two end-points and employ variants of minimal path extraction [14], with improvements such as anisotropic fast matching or anisotropic flow [5, 19], which often suffer from shortcut paths [18]. The high degree of user interactions required and the high false-positive rate in low contrast structures make these approaches inappropriate for our specific task.

Tracking has been used previously in segmentation of vessel structures [12, 17, 23]. Many vessel tracking methods model the state as a cross-sectional ellipse, or a similar shape [4, 12]. However, such a naive representation is inappropriate in our application where the rich background would cause severe leakage of cross-sectional states. In tracking nerve bundles, the regions of low contrast require the state to capture substantially longer segments of the track than what is represented by a cross-section. Other vessel tracking methods use cylindroid [28] states. However, nerves tend to change direction, often sharply, which necessitates a use of more complex descriptors than cylinders. In this work, we introduce a rich state representation that can handle complex tubular structures and variable contrast.

## ■ 2.2 Other Related Work

A recent study has demonstrated the feasibility of using diffusion tensor imaging (DTI) and fiber tracking of lumbar nerves [3] to estimate differences of diffusion parameters between healthy and herniated disks. This work indicates the potential for DTI images to aid in the current segmentation task. Should good quality DTI images become available, our method can be readily extended to include diffusion information by incorporating directional information in the likelihood function.

Recently, the field of connectomics has led to the segmentation of elongated structures in the brain, such as neurites [16, 25]. The general approach is to construct voxel affinity graphs, which are then used to segment structures in the brain volume. However, the task is significantly different than the one addressed in this paper, as even elongated structures are large enough and clearly delineated in the microscopy images used for connectomics, and partial volume plays no significant role.

# Particle Filter

**W**E formulate automatic segmentation of nerves as a tracking problem, where we iteratively track segments, or particles, of tubular bundles using the particle filter. The particles, weighed based on how well they describe the corresponding MRI patch, form a set of point probability masses and represent the likelihood of a nerve at a specific location in the image. For the next iteration, new particles are sampled from this likelihood, propagated onto the next patch, and re-weighed. In this chapter, we review the intuition and mathematical framework of particle filters [1, 15, 26]. In the next chapter, we describe in detail how the various elements of this framework are applied to the task of nerve segmentation.

### ■ 3.1 The Particle Filter

We let  $\mathbf{h}_t$  be a state representation at step  $t$  of the tracking algorithm. The specific details of this representation  $\mathbf{h}_t$  for nerve bundles are provided in the next chapter. We assume that states respect a first order Markov chain, i.e.,

$$p(\mathbf{h}_t|\mathbf{h}_{1:t-1}) = p(\mathbf{h}_t|\mathbf{h}_{t-1}), \quad (3.1)$$

where  $\mathbf{h}_{1:t-1}$  denotes the state history prior to step  $t$ . We let  $\mathbf{z}_t$  be the image-based observation at step  $t$ , which can be interpreted as being generated from state  $\mathbf{h}_t$  via a noisy generative function:  $\mathbf{z}_t = f(\mathbf{h}_t)$ . An example observation  $\mathbf{z}_t$  is illustrated in the next chapter. We assume that given the state, the observations at different time points are independent and depend only on the corresponding state value:

$$p(\mathbf{z}_{1:t}|\mathbf{h}_{1:t}) = p(\mathbf{z}_t|\mathbf{h}_t) \cdot p(\mathbf{z}_{1:t-1}|\mathbf{h}_{1:t-1}). \quad (3.2)$$

At each step  $t$ , the tracking algorithm estimates the posterior distribution  $p(\mathbf{h}_t|\mathbf{z}_{1:t})$

of each state  $\mathbf{h}_t$  given all observations  $\mathbf{z}_{1:t}$ . Starting with Bayes' Rule, and using assumptions (3.1) and (3.2), we find that

$$\begin{aligned}
p(\mathbf{h}_{1:t}|\mathbf{z}_{1:t}) &\propto p(\mathbf{z}_{1:t}|\mathbf{h}_{1:t})p(\mathbf{h}_{1:t}) \\
&= p(\mathbf{z}_t|\mathbf{h}_t)p(\mathbf{z}_{1:t-1}|\mathbf{h}_{1:t-1})p(\mathbf{h}_t|\mathbf{h}_{t-1})p(\mathbf{h}_{1:t-1}) \\
&= p(\mathbf{z}_t|\mathbf{h}_t)p(\mathbf{h}_t|\mathbf{h}_{t-1})p(\mathbf{z}_{1:t-1}, \mathbf{h}_{1:t-1}) \\
&= p(\mathbf{z}_t|\mathbf{h}_t)p(\mathbf{h}_t|\mathbf{h}_{t-1})p(\mathbf{h}_{1:t-1}|\mathbf{z}_{1:t-1})p(\mathbf{z}_{1:t-1}) \\
&\propto p(\mathbf{z}_t|\mathbf{h}_t)p(\mathbf{h}_t|\mathbf{h}_{t-1})p(\mathbf{h}_{1:t-1}|\mathbf{z}_{1:t-1})
\end{aligned}$$

The desired posterior distribution can then be expressed as follows:

$$\begin{aligned}
p(\mathbf{h}_t|\mathbf{z}_{1:t}) &= \sum_{\mathbf{h}_{1:t-1}} p(\mathbf{h}_{1:t}|\mathbf{z}_{1:t}) \\
&\propto p(\mathbf{z}_t|\mathbf{h}_t) \sum_{\mathbf{h}_{1:t-1}} p(\mathbf{h}_t|\mathbf{h}_{t-1})p(\mathbf{h}_{1:t-1}|\mathbf{z}_{1:t-1}) \\
&= p(\mathbf{z}_t|\mathbf{h}_t) \cdot p(\mathbf{h}_t|\mathbf{z}_{1:t-1}).
\end{aligned} \tag{3.3}$$

Therefore, the particle distribution  $p(\mathbf{h}_t|\mathbf{z}_{1:t})$  can be written as a product of a likelihood function  $p(\mathbf{z}_t|\mathbf{h}_t)$  and a prior term  $p(\mathbf{h}_t|\mathbf{z}_{1:t-1})$  [1, 15, 26].

### ■ 3.2 Algorithms

A particle filter maintains the posterior distribution  $p(\mathbf{h}_t|\mathbf{z}_{1:t})$  nonparametrically via a set of samples  $\{\mathbf{h}_t^{(k)}, w_t^{(k)}\}_{k=1}^K$  at step  $t$ , which is generated from the set  $\{\mathbf{h}_{t-1}^{(k)}, w_{t-1}^{(k)}\}_{k=1}^K$  at step  $t-1$ . Specifically, a sample  $\langle \hat{\mathbf{h}}_{t-1}, \hat{w}_{t-1} \rangle$  is drawn from the set  $\{\mathbf{h}_{t-1}^{(k)}, w_{t-1}^{(k)}\}$  – which approximates  $p(\mathbf{h}_{t-1}|\mathbf{z}_{1:t-1})$  – according to the weights  $\{w_{t-1}^{(k)}\}$ . A state vector  $\mathbf{h}_t^{(k)}$  is propagated from  $\hat{\mathbf{h}}_{t-1}$  via the dynamics  $p(\mathbf{h}_t|\mathbf{h}_{t-1})$ . This process constructs a sample from a prior distribution

$$p(\mathbf{h}_t|\mathbf{z}_{1:t-1}) = \sum_{\mathbf{h}_{t-1}} p(\mathbf{h}_t|\mathbf{h}_{t-1})p(\mathbf{h}_{t-1}|\mathbf{z}_{1:t-1}). \tag{3.4}$$

The sample's weight  $w_t^{(k)}$  is set to the likelihood  $p(\mathbf{z}_t|\mathbf{h}_t^{(k)})$ , yielding the final sample  $\langle \mathbf{h}_t^{(k)}, w_t^{(k)} \rangle$ . The particle weights are normalized at each step to sum to 1. The resulting sample set  $\{\mathbf{h}_t^{(k)}, w_t^{(k)}\}_{k=1}^K$  is a representation of the state distribution  $p(\mathbf{h}_t|\mathbf{z}_{1:t})$ , as implied by (3.3).

---

In the next chapter, we define our particle model  $\mathbf{h}_t$  for the nerve bundle, our dynamics model  $p(\mathbf{h}_t|\mathbf{h}_{t-1})$ , and our likelihood measure  $p(\mathbf{z}_t|\mathbf{h}_t)$ . Together with the observations  $\mathbf{z}_t$ , these elements fully define the tracking algorithm.



---

# Nerve Segmentation

**T**HE appearance of nerve tracts in MR images is highly variable, with occasional strong intensity and shape changes due to imaging artefacts, vicinity of other anatomical structures, and nerve damage. This high variability in nerve appearance motivates a statistical representation of the tubular structure. To facilitate such representation, the particle filtering framework considers several hypotheses, or particles, at each tracking iteration, as discussed in the previous chapter. In this chapter, we describe a particle model to characterize the nerve bundles, followed by a dynamics model for the propagation of the particles, and finally the features used in the posterior computation. The framework is summarized in Figure 4.1.

## ■ 4.1 Particle Representation for Nerve Tracks

In our formulation, a particle represents a segment of a nerve bundle. We model each particle  $\mathbf{h}_t$  as a tubular structure around a centerline in 3D, and design the centerline as a Bézier curve [27]. We model the varying thickness of the segment with a quadratic radius function, as illustrated in Figure 4.2. We also maintain the average intensity of the nerve.

A  $n^{\text{th}}$  degree Bézier curve [27] is defined by  $n + 1$  control points. The first and last control points define the end-points of the curve. The interior control points can be thought of as “pulling” the curve towards them. We choose to work with a cubic representation of the segment centerline:

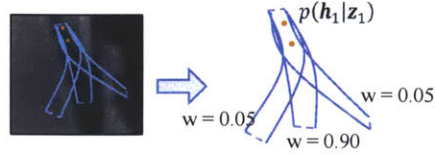
$$\mathbf{c}(\tau) = (1 - \tau)^3 \mathbf{p}_0 + 3(1 - \tau)^2 \tau \mathbf{p}_1 + 3(1 - \tau) \tau^2 \mathbf{p}_2 + \tau^3 \mathbf{p}_3, \quad (4.1)$$

where  $\tau \in [0, 1]$  is the parameterization variable.

We allow the radius function  $r(\cdot)$  to vary quadratically along the segment, and also

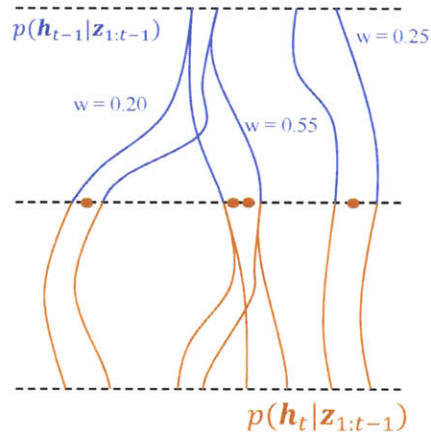
## Nerve Segmentation Algorithm

- Initialize  $\mathbf{h}_0$  with two clicks, and generate first particles  $\{\mathbf{h}_1^k, \mathbf{w}_1^k\}$  representing  $p(\mathbf{h}_1|\mathbf{z}_1)$  from  $\mathbf{h}_0$



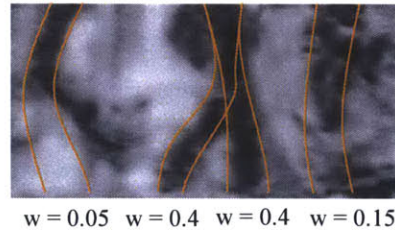
- For successive steps  $2: T$ , until the end of MRI volume:

1. Generate new samples (indicated by  $\bullet$ ) from  $p(\mathbf{h}_{t-1}|\mathbf{z}_{1:t-1})$ .



2. Propagate via  $p(\mathbf{h}_t|\mathbf{h}_{t-1})$  to obtain  $p(\mathbf{h}_t|\mathbf{z}_{1:t-1})$ .

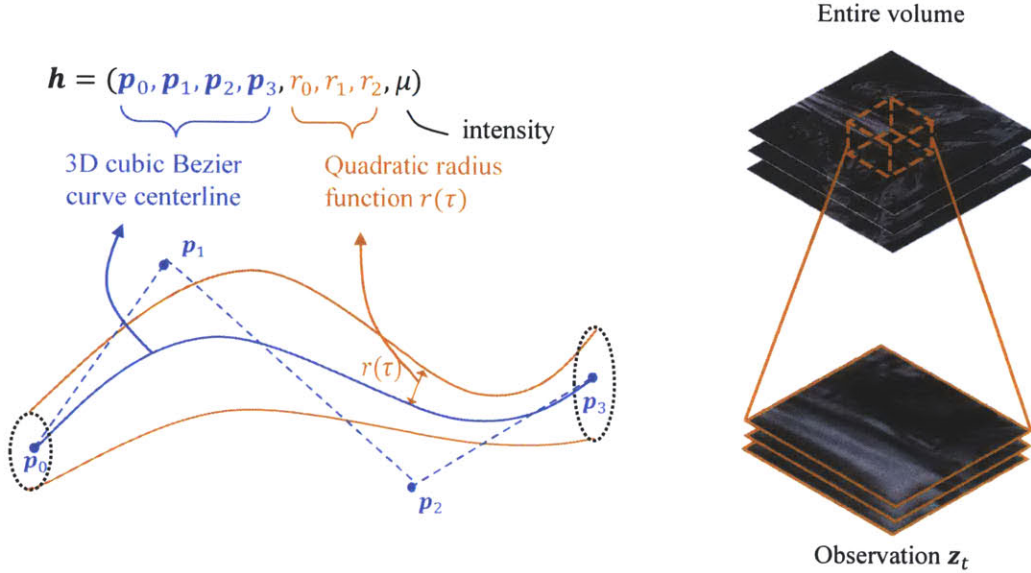
3. Weigh new particles using observations  $\mathbf{z}_t$  to obtain a new particle set  $\{\mathbf{h}_t^k, \mathbf{w}_t^k\}$  and  $p(\mathbf{h}_t|\mathbf{z}_{1:t})$ .



- Clean up and re-score tracks  
Potentially discard tracks if user offers extra click.

**Figure 4.1.** An overview of the Particle Filter algorithm as applied to nerve segmentation. To best illustrate the concepts, we use very few particles in our illustrations, unlike the actual implementation where several thousand particles are used at each step. Following the initialization, we perform the illustrated procedure iteratively. The previous posterior distribution  $p(\mathbf{h}_{t-1}|\mathbf{z}_{1:t-1})$  is approximated by weighed particles (nerve segments). In the first step of an iteration, we obtain samples (orange dots) based on this posterior. In the second step, we propagate these samples, i.e., continue the previous nerve segments, via the dynamics model. Finally, the new particles are weighed based on the current observation with the likelihood model.





**Figure 4.2.** Left: Nerve segment (particle) is defined as a Bézier curve centerline with four control points, a quadratic radius function  $r(\cdot)$ , and the average intensity. Right: An observation is a sub-volume of the entire MRI volume, surrounding a particle hypothesis.

define it via a Bézier curve:

$$r(\tau) = (1 - \tau)^2 r_0 + 2(1 - \tau)\tau r_1 + \tau^2 r_2, \quad (4.2)$$

using control points  $r_0, r_1$ , and  $r_2$ . In addition, we maintain the mean image intensity  $\mu$  inside the segment. The state vector

$$\mathbf{h} = (\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, r_0, r_1, r_2, \mu) \quad (4.3)$$

fully describes the corresponding segment, as illustrated in Figure 4.2. This construction can handle tubular structures with variable directionality, thickness and intensity, such as nerve bundles and ganglia.

## ■ 4.2 Dynamics model

In this section, we describe the dynamics model, which specifies the construction of the current state vector  $\mathbf{h}_t$  given the state vector  $\mathbf{h}_{t-1}$  generated in a previous step of the algorithm. This construction extends the hypothesized nerve ending with  $\mathbf{h}_{t-1}$  with a

new segment  $\mathbf{h}_t$ . As we detail below, we enforce continuity and smoothness between the segments and encourage consistent direction, radius and intensity as the nerve progresses. This procedure represents sampling the probability distribution  $p(\mathbf{h}_t|\mathbf{h}_{t-1})$ .

To ensure continuity of the track, we set the first centerline control point of  $\mathbf{h}_t$  to the last control point of  $\mathbf{h}_{t-1}$ :  $\mathbf{p}_{0,t} = \mathbf{p}_{3,t-1}$ . To maintain smoothness during the transition from the previous to the current particle, we place  $\mathbf{p}_{1,t}$  along the line  $(\mathbf{p}_{2,t-1}\mathbf{p}_{3,t-1})$ . For Bézier curves, the tangent of a curve at an endpoint  $\mathbf{p}_0$  is along the vector  $(\mathbf{p}_0, \mathbf{p}_1)$ , and similarly the tangent at point  $\mathbf{p}_3$  is along the vector  $(\mathbf{p}_2, \mathbf{p}_3)$ . Our construction therefore aligns the tangent vectors of consecutive particles. The distance  $\ell$  between  $\mathbf{p}_{0,t}$  and  $\mathbf{p}_{1,t}$  is drawn uniformly from  $(0, L]$  where  $L$  is a parameter of the algorithm. Formally,

$$\mathbf{p}_{1,t} = \mathbf{p}_{0,t} + \ell \hat{\mathbf{n}},$$

where  $\hat{\mathbf{n}}$  is the unit vector in the direction of  $(\mathbf{p}_{2,t-1}, \mathbf{p}_{3,t-1})$ .

We draw the direction of  $(\mathbf{p}_{1,t}, \mathbf{p}_{2,t})$  from a von Mises-Fisher distribution on the unit sphere, centered on the direction of the initial tangent  $\hat{\mathbf{n}}_{01} = (\mathbf{p}_{0,t}, \mathbf{p}_{1,t})$  with concentration  $1/\sigma_\alpha^2$  and call this new direction  $\hat{\mathbf{n}}_{01} + \Delta\hat{\mathbf{n}}_{12}$ . We then sample the distance  $\ell_2$  of  $\mathbf{p}_{2,t}$  from  $\mathbf{p}_{1,t}$  uniformly from  $(0, L]$ . Formally,

$$\mathbf{p}_{2,t} = \mathbf{p}_{1,t} + \ell_2(\hat{\mathbf{n}}_{01} + \Delta\hat{\mathbf{n}}_{12}).$$

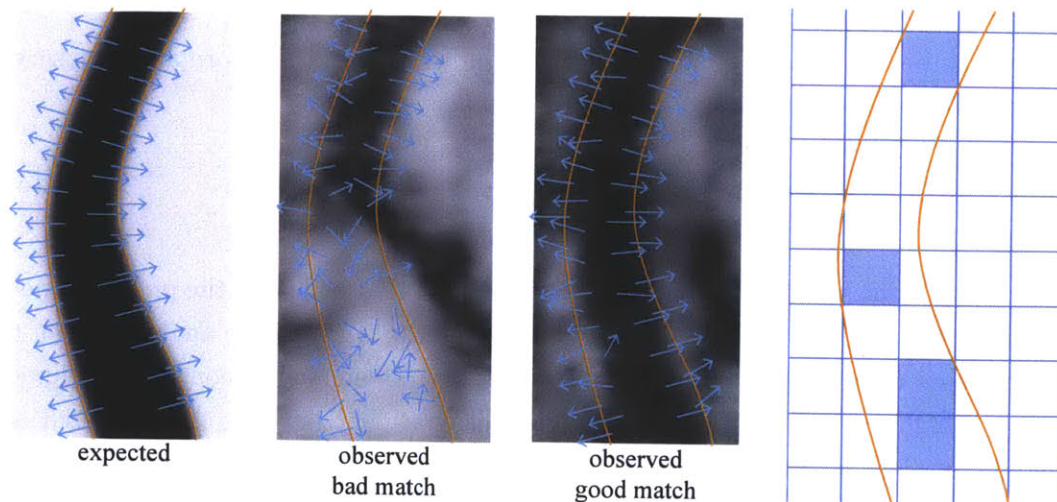
Using the same procedure, we generate  $\mathbf{p}_{3,t}$  based on  $\mathbf{p}_{2,t}$  and vector  $\hat{\mathbf{n}}_{12} = (\mathbf{p}_{1,t}, \mathbf{p}_{2,t})$ , and  $\ell_3 \in [0, L]$ :

$$\mathbf{p}_{3,t} = \mathbf{p}_{2,t} + \ell_3(\hat{\mathbf{n}}_{12} + \Delta\hat{\mathbf{n}}_{23}).$$

Similar continuity and smoothness is enforced on the radius function  $r(\cdot)$ . We set  $r_{0,t} = r_{2,t-1}$  to maintain continuity of the radius function. We sample a distance  $d \in (0, 1)$ , and set  $r_{1,t}$  to the y-coordinate of a control point distance  $d$  away from  $r_{0,t}$  along the line  $(r_{1,t-1}, r_{2,t-1})$ . We choose  $r_{2,t}$  from a Gaussian distribution with mean  $r_{1,t}$  and variance  $\sigma_r^2$ .

The intensity parameter  $\mu_t$  is propagated via a Gaussian distribution with variance  $\sigma_\mu^2$ .

This construction depends on four parameters:  $L$  controls the length of the particle,  $\sigma_\alpha^2$  determines the variation in the particle curvature,  $\sigma_r^2$  describes the range of the radius, and  $\sigma_\mu^2$  captures the variation in intensity. We describe the values for these



**Figure 4.3.** Illustration of likelihood computation and the partial volume effect. Left: hypothesized particle implies gradient directions (blue arrows) in the image. Next to it are two examples of possible observations around this hypothesis. In the first example observation, the proposed intensity is very different from the average intensity over the particle volume, giving rise to high intensity distance  $d_{\mu}^2$ . The gradient directions at each pixel are also very different from the expected directions (at the same voxel), yielding a high gradient distance  $d_{\nabla}^2$  for the first example observation. For the second observed image, both the average intensity over the particle volume and the gradient directions match the expected particle significantly better. Overall, the particle weight will be low given the first observation, but high given the second observation. Right: a thin particle, or nerve segment, is shown where the blue grid defines voxels. Only the shaded voxels are almost entirely in the nerve, the rest are a combination of nerve and background intensities, illustrating the severity of the partial volume effect in our application.

parameters in the next chapter.

### ■ 4.3 Likelihood Measure

Here we focus on the likelihood  $p(\mathbf{z}_t | \mathbf{h}_t)$  of the image patch  $\mathbf{z}_t$  given a particle  $\mathbf{h}_t$ . The contrast between the nerve bundle and surrounding tissue may change along the track, rendering ribbon measures [12, 23], which are often used for this type of task in vessel work, inappropriate. However, image gradient is still typically normal to the centerline in a small neighbourhood around the bundle. Therefore, to measure alignment of an observed image patch  $I$  with a hypothesis  $\mathbf{h}_t$ , we simulate an image patch with a black nerve segment described by the particle  $\mathbf{h}_t$  on a white background as shown in Figure 4.3. We compare the directions of its gradients  $\mathbf{g}^h$  with those of the observed patch  $\mathbf{g}^I$  (blue arrows in Figure 4.3). We avoid using the gradient magnitude, since false hypotheses with partial but very strong contrasts are abundant. We express the

distance  $d_{\nabla}^2$  between the normalized particle gradient  $\hat{\mathbf{g}}^h$  and the normalized observed gradient  $\hat{\mathbf{g}}^I$  via the sigmoid function:

$$d_{\nabla}^2(\mathbf{h}_t, I) = \frac{1}{|V(\mathbf{h}_t)|} \sum_{v \in V(\mathbf{h}_t)} \frac{1}{1 + \exp\left\{-\frac{\theta(\hat{\mathbf{g}}_v^h, \hat{\mathbf{g}}_v^I) - c}{w}\right\}}, \quad (4.4)$$

where  $V(\mathbf{h}_t)$  is the set of voxels in the simulated patch that belong to the predicted nerve segment,  $v$  is an image voxel,  $\theta(\hat{\mathbf{g}}_v^h, \hat{\mathbf{g}}_v^I) = \frac{\pi}{2} - \left| \frac{\pi}{2} - \angle(\hat{\mathbf{g}}_v^h, \hat{\mathbf{g}}_v^I) \right|$  is the angle between the two gradients mirrored around  $\pi/2$ , and  $c, w$  are parameters that control the sigmoid shape. A high gradient distance  $d_{\nabla}^2$  implies disagreement between the expected and observed gradient fields.

To encourage consistent nerve intensity, a second term measures the distance between the mean intensities in the observation and the particle intensity prediction:

$$d_{\mu}^2(\mathbf{h}_t, I) = \left[ \mu_t - \frac{1}{|V(\mathbf{h}_t)|} \sum_{v \in V(\mathbf{h}_t)} I_v \right]^2. \quad (4.5)$$

We form the likelihood model by combining equations (4.4) and (4.5):

$$p(\mathbf{z}_t | \mathbf{h}_t) = \frac{1}{Z} (1 - d_{\nabla}^2) \exp\{-\lambda d_{\mu}^2\}, \quad (4.6)$$

where  $\lambda$  penalizes the intensity distance and  $Z$  is the partition function. In practice,  $Z$  does not need to be explicitly computed as the weights of all samples are normalized at the end of each iteration. In the sigmoid formula (4.4) we use the exponential to transform the gradient angle distance  $\theta(\hat{\mathbf{g}}_v^h, \hat{\mathbf{g}}_v^I)$  to the similarity measure  $1 - d_{\nabla}^2$ , controlled by  $c$  and  $w$ . In (4.6), we similarly use the exponential to obtain a similarity, where the driving parameter is  $\lambda$ .

Partial volume effects play a significant role in this computation. In particular, because nerve bundles are generally thin, most nerve bundle voxels also contain volume from surrounding anatomy (Figure 4.3). This creates a large gradient field as non-negligible gradients appear at most nerve voxels, and very few voxels can be used for mean intensity estimation. Modeling large segments of the nerve tracts, as opposed to cross-sections, improves the robustness of the method by increasing the number of voxels that contribute to the distance computations in (4.4) and (4.5).

## Implementation Details and Parameter Choices

**W**E initialize each nerve bundle with two nearby clicks that specify  $\mathbf{p}_{2,0}$  and  $\mathbf{p}_{3,0}$  for a set of particles of equal weights and varied radii parameter  $r_2$  at time  $t = 0$ . This information fully allows us to start the algorithm loop with  $t = 1$ . We run the particle filter until all bundles reach a pre-determined section of the volume, outside of the spine (such as the end of the volume). In each iteration, we sample many particles but keep only the ten top-weighted particles to form the sample set  $\{\mathbf{h}_t^{(k)}, w_t^{(k)}\}_{k=1}^{10}$ .

To identify the most appropriate of the resulting tracts, we construct each tract  $V(\mathbf{h}_{1:t})$  in the volume, and compute the likelihood (4.6) for the entire tract. At this point, the required nerve is fully segmented, however spurious tracts may also be present in some cases. This might happen if other nerves are in contact with the nerve of interest at some point during tracking, or if structures that seem tubular are near the nerve. A user can provide a single mouse click, such as a point around the ganglia or nerve ends, which eliminates any incorrect tracks.

The parameters described in Section 4.2 are properties of the nerves in our MR images. We found that estimating these parameters from one nerve in one patient provides a good set of parameters for all other images with the same contrast and resolution. We set parameters as follows: length  $L = 15$ , curvature  $\sigma_\alpha = 0.2$ , radius  $\sigma_r = 0.1$ , intensity  $\sigma_\mu = 0.05$ , where the units are all in millimetres.

Parameters  $c, w$  and  $\lambda$  control the shape of the likelihood measure. For the gradient distance measure, we seek to strongly encourage gradients that are close to the predicted ones, and discourage any that are perpendicular to the prediction. Similarly, we strongly discourage any segments whose predicted intensity does not match the observed patch. This selection criteria leads to a choice of  $c = \pi/4$ ,  $w = 0.05$ , and  $\lambda = 50$ .

We typically use 100,000 particles, which leads to runtimes of 5 to 30 CPU-minutes

per nerve in our MATLAB implementation. The particle search is parallelized within each iteration, giving us faster runtimes when utilizing multiple cores. Note that due to the multi-hypothesis nature of particle filters, branching nerves are naturally tracked.

## Results

**W**E evaluate our algorithm on synthetic and patient data. We show that it can fully segment the nerves from the initial input point up to the end of the ganglia, and provide accurate estimates of the nerve thickness. To quantify the accuracy of our method, we measure the distances between the desired and automatic nerve surfaces.

### ■ 6.1 Synthetic Data

First, we generate synthetic nerve tracks via our particle model. We vary the dynamics parameters, such as the radius and control points, beyond ranges that are observed in real images, to allow for tracks with more irregular behavior. The image intensity is then formed by adding white noise to the predicted nerve image:

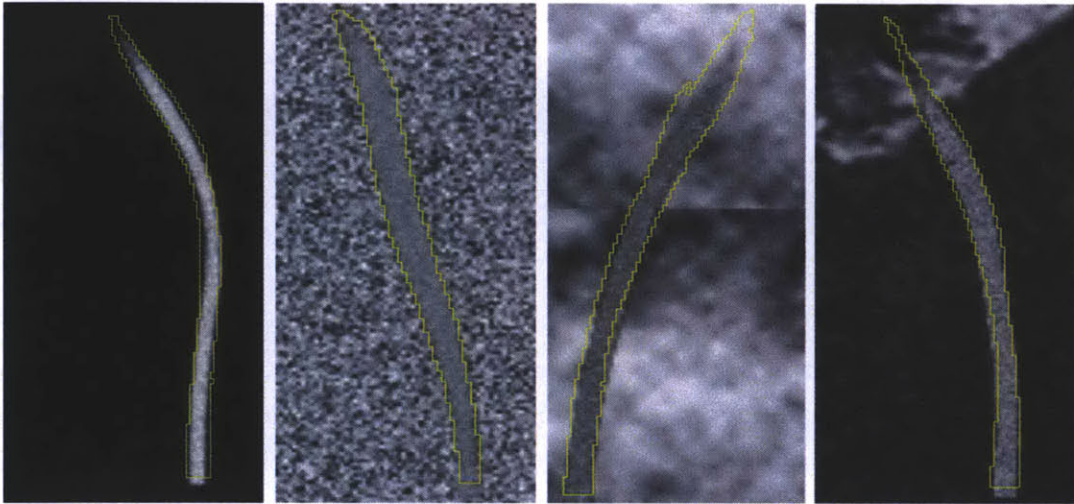
$$I_n(v) \sim \mathcal{N}(0.5n(v), \sigma_n^2), \quad (6.1)$$

where  $n(v) \in [0, 1]$  is the amount of nerve present at voxel  $v$ ,  $\mathcal{N}(\mu, \sigma^2)$  is a Gaussian distribution with mean  $\mu$  and variance  $\sigma^2$ , and  $\sigma_n = 0.05$  is estimated from real MRI scans with manually segmented nerves.

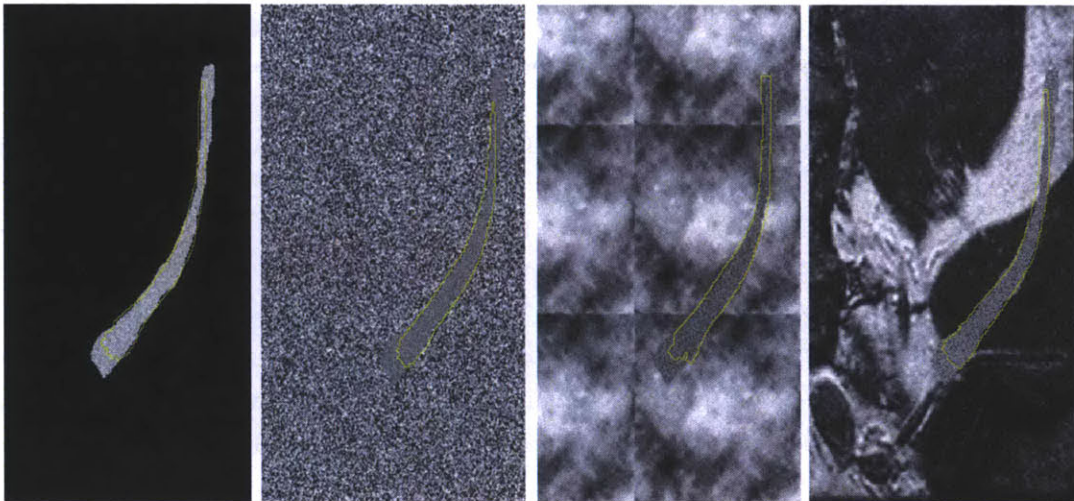
We superimpose the nerve tracks on backgrounds with no noise (blank), Gaussian noise, Perlin noise, and random sections of MRI volumes, which present varying degrees of segmentation difficulty. Backgrounds with Gaussian noise are generated via  $I_b(v) = \mathcal{N}(0.5, \sigma_b^2)$ , where  $\sigma_b = 0.23$  is estimated from image regions surrounding nerves in patient data. We simulate Perlin noise by first combining layers of smaller dimensions with the same Gaussian noise. Each layer is then up-scaled via interpolation to match the size of the original image, and then added together. Our synthetic images are then defined by:

$$I(v) = n(v) \cdot I_n(v) + (1 - n(v)) \cdot I_b(v), \quad (6.2)$$



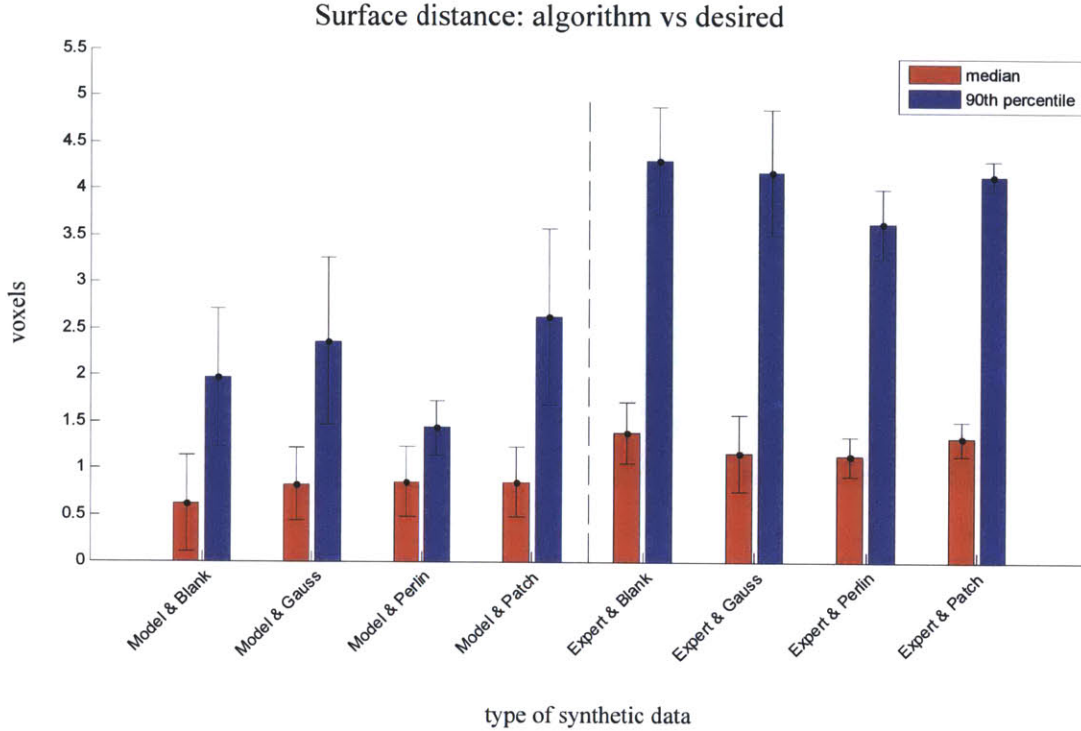


**Figure 6.1.** Synthetic nerves generated from our model with added white noise embedded in different backgrounds: blank, white noise, Perlin noise and anatomical patch, respectively. The yellow outlines show the automatic segmentation results. Slight under-segmentations can be seen around the edges, and occasionally at the ends of the nerve section, where the segmentation has moved into the adjoining slices one voxel too early.



**Figure 6.2.** Real nerve delineation with added white noise and embedded in different backgrounds: blank, white noise, Perlin noise and anatomical patch, respectively. The yellow outlines show automatic segmentations. The segmentations tend to avoid irregularities in the nerves, and will usually under-segment. For such real nerves, note that the segmentations are initialized at the top of the nerves in the shown slice, which leads to under-segmentations in the region due to the initializations.



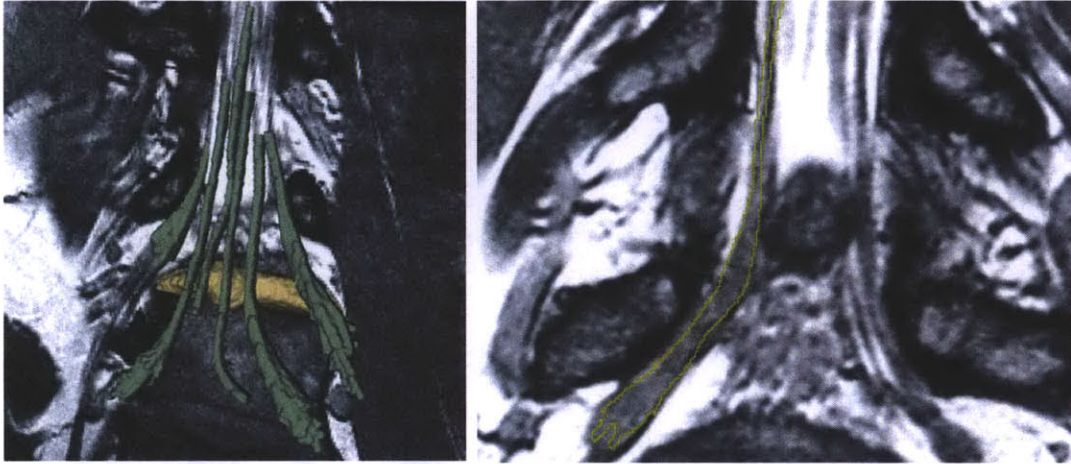


**Figure 6.3.** Summary of synthetic results for the eight types of synthetic nerves and backgrounds. The red bars indicate the average of the median distance between the automatic and the manual outline. The blue bars indicate the 90<sup>th</sup> percentile. For each category, *Model* indicates that we generated the nerve via our particle model, and *Expert* indicates nerves generated from the smoothed expert segmentation of real nerves.

where  $I_n(v)$  is generated via our particle model, and  $I_b(v)$  is one of the four background models. We evaluate the algorithm on 20 images with each background. Example synthetic images and results are shown in Figure 6.1. Typical diameters measure between four and ten voxels, yielding many partial volume voxels between nerve and background.

We also construct and test a second synthetic dataset of the same size, where the initial nerve intensity  $I_n(v)$  is generated from smoothed binary map  $n(v)$  of expert nerve segmentation in patient data. The nerve and background noise models are applied in the same manner as in the first synthetic set. The nerves range from four to 25 voxels in diameter, and present with far more irregular shape, illustrated in Figure 6.2.

As illustrated in Figure 6.3, the median distance between the automatically extracted and true nerve surfaces for each nerve generated from our particle model is  $0.8 \pm 0.4$  voxels, with a 90<sup>th</sup> percentile of  $2.1 \pm 0.7$  voxels. Most of the algorithm errors



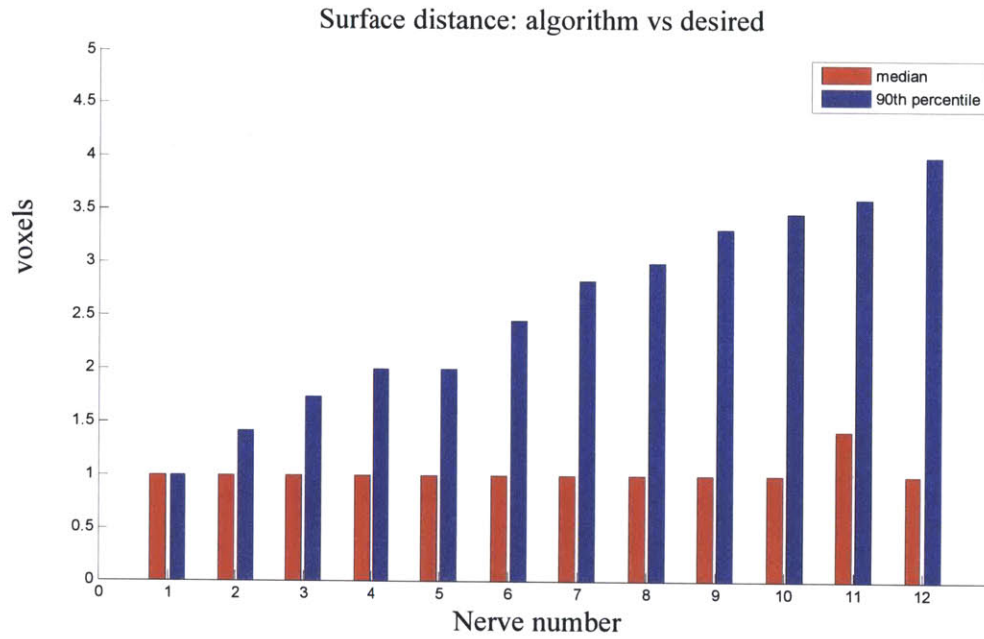
**Figure 6.4.** Patient data results. Left: Rendered segmentation results – a herniated disk (manually segmented, yellow) is impinging on the nerve tracts (segmented with our algorithm with minimal user input, green). Right: Slice that includes a section of a nerve and a ganglion. The yellow outline indicates the automatic delineation of the nerve. The algorithm tends to slightly under-segment the nerve, as we see the margins of the nerve underneath the segmentation.

are in the edge, partial-volume voxels. For nerves generated from expert segmentations, the shape irregularity results in an increased median distance between the automatically extracted and true nerve surfaces of  $1.3 \pm 0.3$  voxels, with the 90<sup>th</sup> percentile of  $4.1 \pm 0.4$  voxels.

## ■ 6.2 Patient Data

We further demonstrate our method on MRI spine scans of 12 nerve bundle segmentations from six subjects. The scans were acquired 3D Wide-band Steady State Free Precession sequence [24] (in-plane resolution 0.44-0.60mm, slice thickness 1.2-1.8mm, TR=6.4-6.9ms, TE=2.1-2.4ms,  $\Theta=25$ ,  $\pm 32$ KHz bandwidth, and FOV=14cm). These include four nerves in two pathologies where the nerves have been displaced by disk herniations. We obtain both expert and automatic segmentations of nerve bundles inside the spine and ganglia that were deemed traceable, and evaluate tracing accuracy for all bundles. Figure 6.4 illustrates an example automatic segmentation.

Quantitative results are summarized in Figure 6.5. The nerve diameters range from three to six voxels inside the spine and up to about 25 voxels in the ganglia. We find that the median distance between the automatically extracted and the expert surface is 1.0 voxel for most nerves, and the 90<sup>th</sup> percentile is  $2.9 \pm 0.6$  voxels.



**Figure 6.5.** Summary of the 12 segmentation results on real patient data. The red bars indicate the medians of distance between the automatic and the expert segmentation; the blue bars indicate the 90<sup>th</sup> percentile, in voxels.



# Discussion

**M**APS showing the general nerve tracks and shapes can be helpful in planning interventions and treatment. Accurate thickness estimation is important as it provides critical diagnostic information, such as the degree of damage due to a herniated disk. Furthermore, quantitative measures of nerve properties enable studies of clinical effects of an injured nerve treatment. The surface distance computed in the previous section is therefore particularly suitable for our application in understanding the quality of thickness estimation.

### ■ 7.1 Algorithm Behavior

In synthetic data without background noise, the algorithm successfully tracks the nerve direction and thickness. The main errors are observed at the nerve edges, where partial volume effects are strong. Gaussian noise background or anatomical data background yield slightly increased surface distances. Perlin noise presents the strongest challenge, as it creates background areas that yield more consistent gradient vectors. For such areas in the vicinity of the nerve, the algorithm may slightly over or under estimate the surface. While for the first three background types no post-processing was necessary, Perlin data sometimes caused the algorithm to estimate spurious tracts in addition to the true one.

In both synthetic and real data, the algorithm fully followed the nerve tracks and provided an accurate estimate of the nerve thickness. Visually, there is only a slight tendency to under-estimate the thickness of the nerve, which is caused by two effects. First, the partial volume effect causes the nerves to have noisy margins. In both datasets, the *true* segmentations (generated ground truth or manual expert segmentations) include these edge voxels, whereas the algorithm tends to omit them as they lower both terms in the likelihood measure (4.6). Depending on the application, the method can

be tuned to expand the segmentation to correct this bias. More importantly, in real data, the nerves and ganglia present with various shapes and irregularities, especially in the context of pathologies. Deformations and protrusions are picked up by expert segmentations, but they do not fit the smooth particle model we impose, and are therefore omitted by the algorithm (Figure 6.4). Severe irregularities are generally more prominent in the area leading up to and including the ganglia where the nerve thickens and is more likely to be affected by neighboring anatomy. Our particle model is therefore flexible enough to capture most of the nerve shape, but does not handle occasionally arbitrary appearance of the nerve bundles. We conclude that the proposed segmentation may slightly under-segment, but provides a solid estimation of the nerve location and thickness. On average, nerves require less than three mouse clicks: two initial, nearby clicks, and potentially one (or very rarely two) post-processing clicks to eliminate spurious branches.

Note that the second synthetic dataset combines the irregularity of real nerves with more complex backgrounds in the case of Perlin noise, which leads to errors from both sources. Here, we observe under-segmentations due to irregular nerve shape and over segmentation due to structure in the Perlin noise. The start of the nerve may also be under-segmented as the initial clicks may not be central in the cross section.

## ■ 7.2 Exceptions

In general, the algorithm cannot succeed on nerves which would otherwise not be segmentable by the user, as not enough contrast is present to distinguish the nerve from other anatomy and guide the likelihood. Finally, we also explored the algorithm on the nerves following the ganglia in patient images, where they split up into several thinner peripheral nerves. Here, the individual fibres lose contrast and often change intensity, and the algorithm is usually only able to segment for no more than a few millimetres. Segmentation of these fibres is left for future work.

# Conclusion

**U**NDERSTANDING the impact on spinal nerves is critical in evaluating spinal pathologies. Creating segmentation maps of the implicated nerves in MR images is an important step in diagnosis and treatment planning, and in aiding of surgical interventions. However, manual segmentation is challenging due to the high variability of nerve contrast and 3D nature of the volume.

### ■ 8.1 Contributions

In this thesis, we proposed and demonstrated the first approach for the segmentation of nerve bundles in high-resolution spine MRI. The method is based on particle filtering and requires minimal input from the user. We model nerve segments via Bézier curves and define a dynamics model for propagating the segments. A new distance measure that utilises gradient fields and nerve intensities is used to score nerve segments and whole bundles.

We evaluate the method's capacity to handle nerves and ganglia in the presence of both high and low contrast in a variety of synthetic and patient data. We demonstrate the ability to fully track the nerve path and estimate the nerve thickness correctly, with small errors in surface estimation. The 3D maps of nerve bundles generated by our algorithm can aid substantially in diagnosis, treatment and intervention planning.

### ■ 8.2 Future Work

The algorithm can be improved by further isolating and segmenting irregularities in the nerve surface. Such a segmentation can be particularly useful in understanding damaged nerve sections. For example, a post processing step can be added which does a local search for irregular nerve shape, perhaps given user input to control leaking into neighboring surfaces. Segmentation of the much thinner peripheral nerves, which also

present with lower contrast, remains a challenging problem for future work.

The segmentation results presented in Chapter 6 facilitate important measures in understanding the effects of a pathology on nerve bundles. Sudden and significant changes in radius or displacement of the nerve indicate likely damage, but subtle changes can also be indicative of pathology. As a continuation of the project, we will be developing a nerve *score* which describes the amount and type of damage done to a nerve bundle. The score will be designed to be informative of symptom duration, treatment outcome, and the need for future surgery. A proposed clinical trial will use this score to differentiate between concordant and discordant pain, helping clinicians understand the source of back and leg pain in complex spine pathologies.



---

---

# Bibliography

- [1] Arulampalam, M.S., Maskell, S., Gordon, N., Clapp T.: A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Process.* 50, 174-188. (2001)
- [2] Aylward, S.R. and Bullitt, E.: Initialization, noise, singularities, and scale in height ridge traversal for tubular object centerline extraction. *IEEE Transactions on Medical Imaging* 21(2), 61-75. (2002)
- [3] Balbi, V., Budzik, J.F., Duhamel, A., Bera-Louville, A., Le Thuc, V., Cotten, A.: Tractography of lumbar nerve roots: initial results. *European Radiology* 21, 1153-1159. (2010)
- [4] Behrens, T., Rohr, K., Stiehl, H.S.: Segmentation of tubular structures in 3d images using a combination of the hough transform and a kalman filter. In: Radig, B., Florczyk, S. (eds.) *Pattern Recognition. LNCS*, vol.2191, pp. 406-413. (2001)
- [5] Benmansour F. and Cohen. L.D.: A new interactive method for coronary arteries segmentation based on tubular anisotropy. In *ISBI: IEEE International Symposium on Biomedical Imaging: 2009*. pp. 41-44. (2009)
- [6] Blume A., Chun W., Kogan D., Kokkevis V., Weber N., Petterson R.W., and Zeiger R.: Google Body: 3D human anatomy in the browser. In *ACM SIGGRAPH 2011 Talks (SIGGRAPH '11)*, ACM, NY, NY: 2011, vol. 19, pp. 1. (2011)
- [7] Boskamp T., Rinck D., Link F., Kummerlen B., Stamm G., Mildenerger P.: New Vessel Analysis Tool for Morphometric Quantification and Visualization of Vessels in CT and MR Imaging Data Sets. *Radiographics.* 24(1), 287-297. (2004)
- [8] Bruijns. J.: Fully-automatic branch labelling of voxel vessel structures. In: *Vision Modeling and Visualization 2001*. pp. 341-350. (2001)

- [9] Cai, W., Dachille, F., Harris G. J., Yoshida, H.: Vesselness propagation: a fast interactive vessel segmentation method In: Proc. SPIE: Society of Photo-Optical Instrumentation. 614447 (2006)
- [10] Dalca A.V., Danagouliau G., Kikinis R.; Schmidt E., Golland P.: Segmentation of Nerve Bundles and Ganglia in Spine MRI using Particle Filters. In: Fichtinger, G., Martel, A.L., Peters, T.M. (eds.) MICCAI: International Conference on Medical Image Computing and Computer Assisted Intervention, 2011, LNCS, vol. 6893, pp 537-545. (2011)
- [11] Delingette H. and Montagnat J.: Shape and topology constraints on parametric active contours. *Computer Vision and Image Understanding* 83, pp. 140-171. (2000)
- [12] Florin, C., Paragios, N., Williams, J.: Particle filters, a quasi-monte carlo solution for segmentation of coronaries. In: Duncan, J.S., Gerig, G. (eds.) MICCAI: International Conference on Medical Image Computing and Computer Assisted Intervention, 2005. LNCS, vol. 3749, pp. 246-253. (2005)
- [13] Fridman, Y., Pizer, S.M., Aylward, S., Bullitt E.: Segmenting 3d branching tubular structures using cores. In: Ellis, R.E., Peters, T.M. (eds.) MICCAI: International Conference on Medical Image Computing and Computer Assisted Intervention, 2003. LNCS, vol.2879, pp. 570-577. (2003)
- [14] Gulsun, M., Tek, H.: Robust vessel tree modeling. In: Metaxas, D., Axel, L., Fichtinger, G., Szekely, G. (eds.) MICCAI: International Conference on Medical Image Computing and Computer Assisted Intervention, 2008. LNCS, vol.5241, pp. 602-611. (2008)
- [15] Fleet D.J. and Jepson. A.D. Foundations of computer vision notes: Tracking, 2009.
- [16] Jain V., Seung H. S., Turaga S. C.: Machines that learn to segment images: a crucial technology for connectomics. *Current Opinion in Neurobiology*, 20(5), pp. 653-666. (2010).
- [17] Lesagea, D., Angelini, E.D., Bloch, I., Funka-Leaa, G.: A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes. *Medical Image Analysis* 13(6), 819-845. (2009)

- [18] Li, H., Yezzi, A.: Vessels as 4-d curves: Global minimal 4-d paths to extract 3-d tubular surfaces and centerlines. *IEEE Transactions on Medical Imaging* 26(9), 1213-1223. (2007)
- [19] Lin, Q.: Enhancement, Detection, and Visualization of 3D Volume Data. Ph.D. Thesis, Dept. EE, Linkoping University, SE-581 83 Linkoping, Sweden, Dissertations No. 824. (May 2003)
- [20] Lorigo, L.M., Faugeras, O.D., Grimson, E.L., Keriven, R., Kikinis, R., Nabavi, A., Westin, C.-F.: Curves: Curve evolution for vessel segmentation. *Medical Image Analysis* 5(3), 195-206. (2001)
- [21] Mille, J., Bone, R., Cohen, L.D.: Region-based 2d deformable generalized cylinder for narrow structures segmentation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV: European Conference on Computer Vision, 2008*. LNCS, vol.5303, pp. 392-404. (2008)
- [22] Mohan V., Sundaramoorthi G., Tannenbaum A.: Tubular surface segmentation for extracting anatomical structures from medical imagery. *IEEE Transactions on Medical Imaging* 29(12), 1945-1958. (2010)
- [23] Schaap, M., Manniesing, R., Smal, I., van Walsum, T., van der Lugt, A., Niessen, W.: Bayesian tracking of tubular structures and its application to carotid arteries in CTA. In: Ayache, N., Ourselin, S., Maeder, A., (eds.) *MICCAI: International Conference on Medical Image Computing and Computer Assisted Intervention, 2007*. LNCS, vol.4792, pp. 562-570. (2007)
- [24] Schmidt, E.J., Shankaranarayanan, A., Jaume, S., Danagouliau, G., Mukundan, S.J., Nayak, K.S.: Wide-band steady state free precession with small diffusion gradients for spine imaging: Application to superior nerve visualization. In: 18th *ISMRM:International Society for Magnetic Resonance in Medicine*, pp. 448. (2010)
- [25] Turaga S. C., Murray J. F., Jain V., Roth F., Helmstaedter M., Briggman K., Denk W., Seung H. S.: Convolutional networks can learn to generate affinity graphs for image segmentation. In: *Neural Computation* 22, pp. 511-538, (2010).
- [26] Gordon, N.J., Salmond, D.J., and Smith, A.F.M.: Novel approach to nonlinear / non-Gaussian Bayesian state estimation. *IEE Proceedings F: Radar and Signal Processing*, 140(2), 107-113. (1993)

- 
- [27] Bartels, R.H., Beatty, J.C., and Barsky, B.A.: *Bézier Curves (Ch. 10): An Introduction to Splines for Use in Computer Graphics and Geometric Modelling*. SF, CA: Morgan Kauf. 211-245. (1998)
- [28] Tyrrell, J., di Tomaso, E., Fuja, D., Tong, R., Kozak, K., Jain R., Roysam B.: Robust 3-d modeling of vasculature imagery using superellipsoids. *IEEE Transactions on Medical Imaging* 26(2), 223-237. (2007)
- [29] Yi, J., Ra, J.B.: A locally adaptive region growing algorithm for vascular segmentation. *International Journal of Imaging Systems and Technology*, 13(4), 208-214. (2003)